

(19)



Europäisches Patentamt  
European Patent Office  
Office européen des brevets



(11) Publication number:

**0 598 995 A1**

(12)

**EUROPEAN PATENT APPLICATION**

(21) Application number: 93113923.2

(51) Int. Cl.5: H04N 1/41

(22) Date of filing: 31.08.93

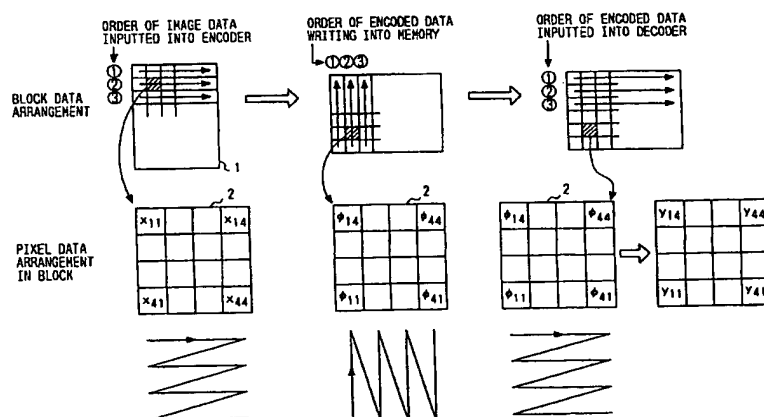
(30) Priority: 26.11.92 JP 316759/92

(43) Date of publication of application:  
01.06.94 Bulletin 94/22(64) Designated Contracting States:  
DE FR GB(71) Applicant: MITSUBISHI DENKI KABUSHIKI  
KAISHA  
2-3, Marunouchi 2-chome  
Chiyoda-ku  
Tokyo(JP)(72) Inventor: Oka, Kenichiro, c/o Mitsubishi Denki  
K.K.Fukuyama Seisakusho,  
1-8 Midori-machi  
Fukuyama-shi, Hiroshima(JP)(74) Representative: KUHNEN, WACKER &  
PARTNERAlois-Stelnecker-Strasse 22  
D-85354 Freising (DE)

(54) Image encoding method and image encoding/decoding method.

(57) An encoding method of a GBTC type that encoded data and decoded data have a fixed length, which can reduce the degree of deterioration of the quality of a reproduced image and also to provide an encoding/decoding method which, in encoding or decoding, is capable of performing editing processings to rotate an original image in 90° and to rotate it inversely with respect to up and down, right and left. When encoded data in each of blocks of an original image and level specification signals  $\phi_{ij}$  of respective pixels in each block are written into a memory or are read out therefrom, the fractions of parameter values are rounded to the nearest whole number or cut away to thereby put operation values into integral numbers, or the encoded data and the pixel level specification signal are arranged in such a manner that they can be rotated or inversely rotated an integral number of times 90°.

FIG. 3



EP 0 598 995 A1

## BACKGROUND OF THE INVENTION

The present invention relates to an image encoding method which is suitable for processing an image having multi-tone gradations as well as an image encoding/decoding method which is capable of editing an image while it is encoded data.

As a conventional block truncation encoding method which divides image data into a plurality of blocks each consisting of a plurality of pixels and approximates the image data by a small number of representative gradations in each of the blocks, for example, there is known a method which is disclosed in Unexamined Japanese Patent Publication (Kokai) Hei-1-188166.

The procedures of the conventional encoding method are as follows:

(1) An image is divided into a plurality of blocks each consisting of a plurality of pixels.

(2) In each block, there is found a difference  $D$  between the maximum value of gradation levels  $L_{\max}$  and the minimum value thereof  $L_{\min}$ , that is,  $D = L_{\max} - L_{\min}$ , and the resultant differences  $D$  are classified into the following three kinds of encoding modes:

Mode A: when  $D$  is small;

Pixels in the block are quantized to 1 level.

Mode B: when  $D$  is comparatively large;

Pixels in the block are quantized to 2 level.

Mode C: when  $D$  is large;

pixels in the block are quantized to 4 level.

(3) When the quantized level is 1 level, then an average value is used; when 2 level, then the respective representative values of a pixel group larger than the average value and a pixel group smaller than the average value are used; and, when 4 level, then 4 values at regular distances and corresponding to gradation distributions in the block are used.

(4) Each of the blocks is designated by a reference level  $L_a$  for regulating the quantization level in the block, a level distance  $L_d$ , and a level specifying signal (2 bits/pixel) for regulating a quantization level for every pixel.

(5) The level specifying signal is connected between the blocks to be converted into two bit plane images ( $\phi_1, \phi_2$ ), and are then encoded by use of binary image codes such as MMR (CCITT/T. 6) or the like, respectively. Also, the level distance  $L_d$  is encoded by use of a variable length code, and the reference level  $L_a$  encodes a difference between the blocks by use of a variable length code.

Also, according to the conventional encoding method, there are employed the following several means in order to enhance an encoding efficiency:

(1) The encoding modes are classified into 3 kinds according to the differences between the maximum and minimum values of the gradation level of the pixels in each of the blocks to change the quantizing level number.

(2) The level specifying signal is converted into the two bit plane images and the two bit plane images are respectively encoded in terms of codes for binary images.

(3) The level distance  $L_d$  is encoded in terms of a variable length code.

(4) The reference level  $L_a$  encodes a difference between the blocks by use of a variable length code.

In the variable length codes encoded in a block unit in this manner, the code data length thereof varies in the respective blocks. Therefore, it is difficult to take out an arbitrary block from the code data group for the purpose of image editing. Further, due to the fact that the reference level  $L_a$  is encoded by a difference between the blocks in a variable length code, even if only the codes in an arbitrary block are taken out, they cannot be decoded but the codes must be decoded sequentially from the first block in the same order as they are encoded.

As an encoding method to solve the above problems, there is known an encoding method of a GBTC (Generalized Block Truncation Coding) type which is disclosed in the draft of a meeting for reading papers of the Institute of Image Electronics Engineers of Japan, "evaluation of an image compression method in a hard copy device" 1991-04-01. Description will be given below of this encoding method.

In Fig. 5, there is shown an explanatory view of the above encoding method of a GBTC type. In Fig. 5, reference character 1 designates an original image; 2, a block which is composed of a square consisting of  $4 \times 4$  pixels;  $X_{ij}$  ( $i, j = 1, 2, 3, 4$ ), a piece of pixel data (which will be hereinafter called " $X_{ij}$ "); and  $\phi_{ij}$  ( $i, j = 1, 2, 3, 4$ ), a level specification signal (which will be hereinafter referred to as " $\phi_{ij}$ ").

In Tables 1 and 2, there are shown an algorithm of a conventional GBTC type encoding method in which encoded data respectively have a fixed length.

[TABLE 1]

GBTC type encoding algorithm	
5	$P1 = (Lmax + 3Lmin)/4$ $P2 = (3Lmax + Lmin)/4$ $Q1 = \text{mean value of all } X_{ij} (X_{ij} \leq P1)$ $Q4 = \text{mean value of all } X_{ij} (X_{ij} > P2)$ $La = (Q1 + Q4)/2$ 10 $Ld = Q4 - Q1$ $L1 = La - Ld/4$ $L2 = La + Ld/4$ for (i = 1, 2, 3, 4) for (i = 1, 2, 3, 4) 15 if $X_{ij} \leq L1$ $\phi_{ij} = 01$ (binary) else if $X_{ij} \leq La$ $\phi_{ij} = 00$ (binary) else if $X_{ij} \leq L2$ $\phi_{ij} = 10$ (binary) else $\phi_{ij} = 11$ (binary) end if end for 20 end for

25

[TABLE 2]

GBTC type decoding algorithm	
30	for (i = 1, 2, 3, 4) for (i = 1, 2, 3, 4) if $\phi_{ij} = 01$ $Y_{ij} = La - Ld/2$ else if $\phi_{ij} = 00$ $Y_{ij} = La - Ld/6$ else if $\phi_{ij} = 10$ $Y_{ij} = La + Ld/6$ 35 else $Y_{ij} = La + Ld/2$ end if end for end for

40

Fig. 6 is a block circuit diagram of a signal process system using the above encoding method, in which reference numeral 3 designates a host device; 4, an encoding circuit; 5, a decoding circuit 5; and 6, an image data memory (which will be hereinafter called a memory).

Also, Fig. 7 is a block circuit diagram of the encoding circuit 4, in which reference numeral 7 designates a buffer memory; 8, an encoding operation circuit; 9, a reference level buffer; 10, a level interval buffer; 11, a level specification signal buffer; and 12, a signal control circuit.

Next, description of the encoding method will be given below. In the conventional encoding method, it is assumed that the pixel data  $X_{ij}$  is composed of 1 byte. The original image 1 is divided into blocks 2 each consisting of  $4 \times 4$  pixels, and each of the blocks is then encoded. That is, at first, it is divided between the maximum value  $Lmax$  and minimum value  $Lmin$  pixels in each block 2 into four equal parts, in which at quarter value from the bottom is expressed as  $P1$  while a quarter value from the top is expressed as  $P2$ . Next, the average value of pixels values ranging from  $Lmin$  up to  $P1$  is expressed as  $Q1$ , while the mean value of pixel values ranging from  $Lmax$  down to  $P2$  is expressed as  $Q4$ .

Then, the reference level  $La$  of the block is found according to the following equation:

55

$$La = (Q1 + Q4)/2$$

And, the level interval  $Ld$  is found according to the following equation:

$$L_d = Q_4 - Q_1$$

After then, the level interval  $L_d$  is divided into four equal parts, in which a quarter value from the bottom is expressed as  $L_1$  while a quarter value from the top is expressed as  $L_2$ . Next, with  $L_1$ ,  $L_a$  and  $L_2$  as the threshold values thereof, 16 pixels within the block are respectively quantized in the following manner:

When  $L_{\min} \leq x_{ij} \leq L_1$ ,  $\phi_{ij} = 01$  (binary value);

When  $L_1 < x_{ij} \leq L_a$ ,  $\phi_{ij} = 00$  (binary value);

When  $L_a < x_{ij} \leq L_2$ ,  $\phi_{ij} = 10$  (binary value);

When  $L_2 < x_{ij} \leq L_{\max}$ ,  $\phi_{ij} = 11$  (binary value).

Now, description will be given below of the operation of the encoding circuit 4 with reference to Figs. 7 and 8. In Fig. 7, the image data corresponding to 4 lines is once stored in the buffer memory 7 and are then transmitted to the encoding operation circuit 8 by blocks, in which  $L_a$ ,  $L_d$ , and  $\phi_{11}$  to  $\phi_{44}$  are found according to such a procedure as described above. The thus found  $L_a$ ,  $L_d$  and  $\phi_{11}$  to  $\phi_{44}$  are then sent respectively through the reference level buffer 9, level interval buffer 10 and level specification signal buffer 11 to the signal control circuit 12, in which they are collected as code data by blocks as shown in Fig. 8.

The code data is written in the memory 6 in the form of a code system arranged in such a manner as shown in Fig. 8. If 1 byte is allocated to  $L_a$ , 1 byte is allocated to  $L_d$ , and 2 bits are allocated to each of  $\phi_{11}$  to  $\phi_{44}$ , then the length of the encoded data belonging to 1 block is 6 bytes. In this manner, the code length of each block is a fixed length of 6 bytes, so that it is simple to take out the code data of an arbitrary block from the code system.

Fig. 9 is a block diagram of the decoding circuit 5 shown in Fig. 6, in which reference numeral 14 stands for a signal distribution circuit; 15, a reference level buffer; 16, a level interval buffer; 17, a level specification signal buffer; 18, a decoding operation circuit; and 19, a buffer memory, respectively.

Fig. 10 is an explanatory view of a conventional GBTC type decoding method shown in Table 2, in which reference character  $y_{ij}$  ( $i, j = 1, 2, 3, 4$ ) stands for decoded pixel data.

Next, the conventional decoding method will be described below. In this method, as shown in Fig. 10, a piece of encoded data is composed of  $L_a$ ,  $L_d$ , and  $\phi_{ij}$ . On the basis of these parameters, the following operation is performed to thereby find the decoded pixel data  $y_{ij}$ .

When  $\phi_{ij} = 01$ , then  $y_{ij} = L_a - L_d/2$

When  $\phi_{ij} = 00$ , then  $y_{ij} = L_a - L_d/6$

When  $\phi_{ij} = 10$ , then  $y_{ij} = L_a + L_d/6$

When  $\phi_{ij} = 11$ , then  $y_{ij} = L_a + L_d/2$

Next, description will be given below of the operation of the decoding circuit 5. In Fig. 9, encoded data in 1 block are divided by the signal distribution circuit 14 into  $L_a$ ,  $L_d$ , and  $\phi_{11}$  to  $\phi_{44}$ , the three kinds of signals are respectively distributed to the reference level buffer 15, level interval buffer 16, and level specification signal buffer 17, the decoded pixel data  $y_{ij}$  is computed by the decoding operation circuit 18, the decoded pixel data  $y_{ij}$  is written into the buffer memory 19 every block, and the decoded pixel data  $y_{ij}$  is outputted every 4 lines.

In the conventional GBTC type encoding method employing a fixed code length, there are produced fractions in an operation to find the respective parameters  $P_1$ ,  $P_2$ ,  $L_a$ ,  $L_d$ ,  $L_1$  and  $L_2$  for use in encoding and in an operation to find the decoded pixel data  $y_{ij}$  for use in decoding. For this reason, when the conventional method is used to repeat encoding and decoding processings on the same image data, the encoded data and decoded data continue to vary each time the processings are performed, with the result that, when the data are reproduced as an image, the quality thereof continues to deteriorate.

#### SUMMARY OF THE INVENTION

The present invention aims at eliminating the above problems found in the conventional GBTC type encoding method. Accordingly, it is an object of the invention to provide a GBTC type encoding method employing a fixed code length which can restrict the deterioration of the quality of the reproduced image from an original image to a small range even if encoding and decoding processings are performed repeatedly.

It is another object of the invention to provide an encoding/decoding method which, in encoding or decoding, is able to perform an editing processing to rotate an image in  $90^\circ$  unit or an editing processing to turn the image upside down or turn inversely the right and left of the image.

The image encoding method of the invention is an encoding method of a fixed length GBTC type, in which fractions to be produced in an operation to encode or decode an image are rounded to the nearest

whole number or cut away to thereby arrange all parameters and signals into integral numbers.

In the image encoding/decoding method according to the invention, in encoding, the encoded data in each of the blocks and the level specification signals of the respective pixels in each of the blocks are arranged in a memory in such a manner that they can be respectively rotated forwardly or inversely by an integral number of times  $90^\circ$  according to cases.

Also, in decoding, when the encoded data of the respective blocks and the level specification signals of the respective pixels in each of the blocks are taken out from the memory, they are respectively read out so that they can be arranged in such a manner that they can be rotated forwardly or inversely an integral number of times  $90^\circ$ .

According to the image encoding method of the invention, due to the fact that fractions to be produced in the encoding and decoding operations are rounded to the nearest whole number or cut away to thereby arrange the respective parameters and the level specification signals of the respective pixels in each of the blocks into integral numbers, when encoding and decoding processings are performed repeatedly on the same image data, encoded data and decoded data are converged at the third encoding or decoding processing at the latest, respectively.

Also, in the image encoding/decoding method of the invention, in encoding, the code data and the level specification signals in the respective blocks are arranged in such a manner that they can be rotated forwardly or inversely an integral number of times  $90^\circ$  according to cases, which makes it possible to edit an image in encoding.

Further, in decoding, the code data and the level specification signals in the respective blocks are arranged in such a manner that they can be rotated forwardly or inversely an integral number of times  $90^\circ$  according to cases, which makes it possible to edit an image in decoding.

#### BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is an explanatory view of a case in which the coded data reaches its converged condition at the third encoding processing in an encoding method according to the embodiment 1 of the invention;  
 Fig. 2 is an explanatory view of a case in which the decoded data reaches its converged condition at the second decoding processing in an encoding method according to the embodiment 2 of the invention;  
 Fig. 3 is an explanatory view of a case in which an image editing processing is executed at the encoding stage of the embodiment 2 of the invention;  
 Fig. 4 is an explanatory view of a case in which an image editing processing is executed at the decoding stage of the embodiment 3 of the invention;  
 Fig. 5 is an explanatory view of an conventional encoding method of a fixed length GBTC type;  
 Fig. 6 is a block diagram of an encoding/decoding circuit using the conventional encoding method of a fixed length GBTC type;  
 Fig. 7 is a block diagram of an encoding circuit using an encoding method of a fixed length GBTC type;  
 Fig. 8 is an explanatory view of the arrangement of the respective parameters and level specification signals as well as the code system in reading according to the encoding method of a fixed length GBTC type;  
 Fig. 9 is a block diagram of a decoding circuit using an encoding method of a fixed length GBTC type; and,  
 Fig. 10 is an explanatory view of a decoding method of a fixed length GBTC type.

#### DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

##### Embodiment 1

In Tables 3 and 4, there is shown an algorithm employed in the GBTC type encoding method of the invention.

[TABLE 3]

5

## GBTC type encoding algorithm in Embodiment

10

$P1 = (L_{max} + 3L_{min})/4$  (fractions to be rounded to the nearest whole number or cut away)

$P2 = (3L_{max} + L_{min})/4$  (fractions to be rounded to the nearest whole number or cut away)

15

$Q1 = \text{mean value of all } X_{ij} (X_{ij} \leq P1)$   
(fractions to be rounded to the nearest whole number or cut away)

$Q4 = \text{mean value of all } X_{ij} (X_{ij} \geq P2)$   
(fractions to be rounded to the nearest whole number or cut away)

20

$La = (Q1 + Q4)/2$  (fractions to be rounded to the nearest whole number or cut away)

$Ld = Q4 - Q1$

25

$L1 = La - Ld/4$  (fractions to be rounded to the nearest whole number or cut away)

$L2 = La + Ld/4$  (fractions to be rounded to the nearest whole number or cut away)

for (i = 1, 2, 3, 4)

30

for (i = 1, 2, 3, 4)

if  $X_{ij} \leq L1$   $\phi_{ij} = 01$  (binary)

else if  $X_{ij} \leq La$   $\phi_{ij} = 00$  (binary)

else if  $X_{ij} \leq L2$   $\phi_{ij} = 10$  (binary)

else  $\phi_{ij} = 11$  (binary)

35

end if

end for

end for

40

45

50

55

[TABLE 4]

	GBTC type decoding algorithm
5	
	for (i = 1, 2, 3, 4)
	for (i = 1, 2, 3, 4)
	if $\phi_{ij} = 01$ $Y_{ij} = La - Ld/2$
10	(fractions to be rounded to the
	nearest whole number or cut away)
	else if $\phi_{ij} = 00$ $Y_{ij} = La - Ld/6$
	(fractions to be rounded to the
	nearest whole number or cut away)
15	else if $\phi_{ij} = 10$ $Y_{ij} = La + Ld/6$
	(fractions to be rounded to the
	nearest whole number or cut away)
	else $Y_{ij} = La + Ld/2$
20	(fractions to be rounded to the
	nearest whole number or cut away)
	end if
	end for
25	end for

The algorithm of the present encoding method is different from the algorithm of the conventional encoding method shown in Tables 1 and 2 in that Q4 is the mean value of pixel values which range from Lmax down to P2 and also that, when there are produced fractions in the operation values used to find the parameters P1, P2, Q1, Q4, La and yij, the fractions are rounded to the nearest whole number or cut away to thereby arrange the respective values into integral numbers respectively. Such integral number arrangement processings are performed by means of the encoding circuit and decoding circuit.

In Fig. 1, there is shown an example in which, when the encoding and decoding processings are repeatedly performed on the same image data in the encoding method according to the embodiment 1, the image data reaches a converged condition. In Fig. 1, reference characters 24a and 24b designate encoded data; 25a and 25b, decoded image; 26, converged encoded data; and 27, a converged decoded image, respectively. In this case, at first the encoded data 26 is converged and then the decoded image 27 is also converged. After once converged, even if encoding and decoding processings are performed again on the same image data repeatedly, no further quality deterioration will occur at all. For this reason, after converged, the present encoding method is substantially equivalent to a reversible encoding method.

In Fig. 2, there is shown an example in which the image data reaches its converged condition at the second decoding processing. In this case, at first the decoded image 27 is converged and then the encoded data is also converged.

#### Embodiment 2.

In Fig. 3, there is shown another embodiment, namely, embodiment 2 of an image encoding/decoding method according to the invention in which an operation of editing an image is executed at an encoding stage, specifically in this case, by rotating the image to the left 90°. In writing the encoded data into the memory 6, the encoded data are not written in the order that they are inputted, but the data are rearranged in such a manner that the 90° left rotation editing is included. In particular, the mutual data arrangement between blocks is made in such a manner as shown in Fig. 3. That is, in writing into the memory 6 the encoded data of an original image that are sequentially inputted by scanning the image from left top to the right direction, from top down to bottom, the data are rearranged sequentially by scanning the image from left bottom to the upper left direction, from left to right, before they are written into the memory 6. Similarly,  $\phi_{ij}$  are rearranged in such a manner as shown in Fig. 3. When decoding the encoded data arranged in the above manner, if they are decoded in a normal manner, that is, if they are read out sequentially by

scanning the image from left top to the right direction, from top to bottom, then the reproduced image is rotated left  $90^\circ$  relative to its original image.

To rotate the original image left  $180^\circ$ , the figures of Fig. 3 may be rotated left further  $90^\circ$  and, in this arrangement, the block data and pixel data may be written into the memory 6 and then the data may be read out in the same order as shown in the right-most figures. And, to rotate the original image left  $270^\circ$ , the figures of Fig. 3 may be rotated left  $180^\circ$ .

In the embodiment 2, there is shown a case in which an image is rotated left an integral number of times  $90^\circ$ . However, the invention is not limited to this, but it can similarly apply to other cases in which the image is rotated right or is turned upside down or vice versa. Further, the invention can similarly apply to an editing processing which includes a combination of the image right and left rotation with the up-and-down and right-and-left inversed rotations.

### Embodiment 3

In Fig. 4, there is shown a still further embodiment, that is, embodiment 3 according to the invention, in which an image editing processing is executed at a decoding stage. Specifically, in the embodiment 3, there is shown a case in which an original image is rotated left  $90^\circ$ . In the embodiment 3, when reading out the encoded data from the memory 6, the encoded data are not read out in the order that they have been written there, but the data are to be rearranged in such a manner that a  $90^\circ$  left rotation editing can be included. Concretely, the mutual data arrangement between the blocks is changed in such a manner as shown in Fig. 4. That is, in writing, the encoded data are inputted in the order that the image is scanned sequentially from left top to the right direction, from top to down, while in reading the data are rearranged by scanning sequentially from right top to the downward direction, from right to left. The pixel data  $\phi_{ij}$  are similarly rearranged as shown in Fig. 4. If the thus read-out encoded data are decoded, then the reproduced image is found that it is rotated  $90^\circ$  with respect to its original image.

To rotate an original image left  $180^\circ$ ,  $270^\circ$ , and to rotate it inversely with respect to up and down and right and left, the data of the original image may be read out from the memory 6 by changing the image scanning order.

In the above-mentioned embodiments, each of the blocks is considered as a square having  $4 \times 4$  pixels. However, the shape and number of pixels of each block are not limited to this. In this case, the buffer memory 7 for storing the data before encoded and the buffer memory 19 for storing the data after encoded may be respectively constructed such that they have the number of lines that matches the size of the block.

Also, although in the embodiments, the amount of data of 1 pixel is 8 bits, the data amount may be different. Further, the reference level  $L_a$  and level interval  $L_d$  are respectively allocated 1 byte and level specification signals  $\phi_{11}$  to  $\phi_{44}$  are respectively allocated 2 bits. However, other different arrangements may be employed for fixing the length of the codes.

To the memory 6 in the encoding/decoding device shown in Fig. 6, there can be applied a secondary storage medium such as a semiconductor memory, a magnetic disk or the like. Also, as a system which uses the present encoding method, there is available an image data base or the like.

In Fig. 6, the encoding circuit 4 and decoding circuit 5 are disposed externally of the memory 6. However, they can be incorporated integrally in the memory 6. Also, both of the encoding circuit 4 and decoding circuit 5 may be incorporated in the host device 3.

In the embodiments 2 and 3, there is shown a case in which an image is rotated left an integral number of times  $90^\circ$ . However, the invention is not limited to this but it can similarly apply to right rotation, up-and-down inverted rotation, right-and-left inverted rotation and the like. Further, the invention can similarly apply to an editing processing which combines the right and left rotation with the up-and-down and right-and-left inverted rotations.

As has been described heretofore, according to the invention, there is provided an image encoding method in which the data length is fixed with respect to the reference level signal, level interval signal and level specification signal and, as the whole of the codes, every block has a fixed length. In the present image encoding method, due to the fact that the fractions of the operation values of the respective parameters for use in encoding as well as the fractions of the level specification signals values of the respective pixels in the blocks calculated in decoding are rounded to the nearest whole number or cut away, when encoding and decoding processings are performed repeatedly on the same image data, the encoded data and decoded data are respectively converged at the third encoding or decoding processing at latest. Thanks to this, even if the encoding and decoding processings are performed repeatedly, the quality of the reproduced image can be restricted to a small range of deterioration when compared with the quality of the original image.



Also, according to the image encoding/decoding method of the invention, in encoding, the coded data in the respective blocks and the level specification signals of the respective pixels in the respective blocks to be written into the memory are arranged such that the coded data and specification signals can be respectively rotated or inversely rotated an integral number of times  $90^\circ$ , which makes it possible to edit an image in encoding.

Further, in decoding, the encoded data in the respective blocks and the level specification signals of the respective pixels in the respective blocks written into the memory in the order of generation are read out in a scanning pattern which, according to cases, allows the data and signals to be rotated or inversely rotated an integral number of times  $90^\circ$ , so that it is possible to edit an image in decoding.

## Claims

### 1. An image encoding method comprising the steps of:

dividing an original image into a plurality of blocks and setting representative gradation levels respectively representing the gradation levels of the respective pixels in each of said blocks;  
calculating reference levels for the respective blocks from the representative gradation levels of the respective pixels;  
setting level intervals obtained by classifying the distribution range of the representative gradation levels of the respective pixels in each of said blocks into a given number;  
creating level specification signals respectively indicating to which one of said set classes each of said representative gradation levels of said respective blocks belongs;  
encoding said reference level signals, level interval signals and level specification signals into data of a fixed length independently of one another; and  
calculating decoded pixel data from said reference level signals, level interval signals and level specification signals to decode said encoded data,  
when said reference levels and level intervals are calculated and when said decoded pixel data are calculated, there is performed an operation to round fractions to the nearest whole number to thereby arrange the calculated values as integral numbers.

2. An image encoding method as claimed in claim 1, wherein said operation to round fractions is one of a cut-off operation and a half-adjust operation.

### 3. An image encoding/decoding method comprising the steps of:

dividing an original image into a plurality of blocks and setting representative gradation levels respectively representing the gradation levels of the respective pixels in each of said blocks;  
calculating reference levels for the respective blocks from the representative gradation levels of the respective pixels;  
setting level intervals obtained by classifying the distribution range of the representative gradation levels of the respective pixels in each of said blocks into a given number;  
creating level specification signals respectively indicating to which one of said set classes each of said representative gradation levels of said respective blocks belongs;  
encoding said reference level signals, level interval signals and level specification signals into data of a fixed length independently of one another; and  
calculating decoded pixel data from said reference level signals, level interval signals and level specification signals,  
when said encoded data in each of said blocks and said level specification signals of the respective pixels in each of said blocks are read out from said original image as pixel data, said data and signals are written into a memory in such a scanning order that they can be rotated or inversely rotated an integral of times  $90^\circ$  according to cases.

### 4. An image encoding/decoding method comprising the steps of:

dividing an original image into a plurality of blocks and setting representative gradation levels respectively representing the gradation levels of the respective pixels in each of said blocks;  
calculating reference levels for the respective blocks from the representative gradation levels of the respective pixels;  
setting level intervals obtained by classifying the distribution range of the representative gradation levels of the respective pixels in each of said blocks into a given number;  
creating level specification signals respectively indicating to which one of said set classes each of

EP 0 598 995 A1

said representative gradation levels of said respective blocks belongs;

encoding said reference level signals, level interval signals and level specification signals into data of a fixed length independently of one another; and

calculating decoded pixel data from said reference level signals, level interval signals and level specification signals,

when said encoded data of said respective blocks and said level specification signals of the respective pixels in said respective blocks are read out from a memory in which said encoded data and said specification signals are written, they are read out in such a scanning order that they can be rotated or inversely rotated an integral number of times 90° according to cases.

FIG. 1

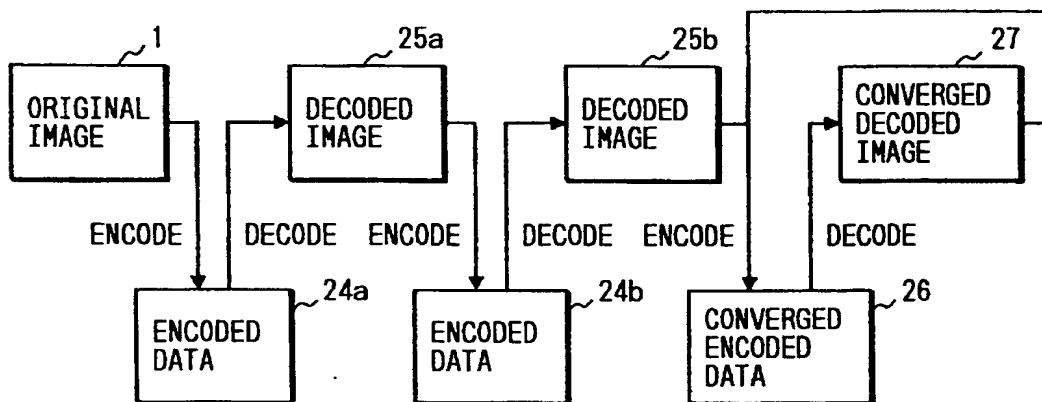


FIG. 2

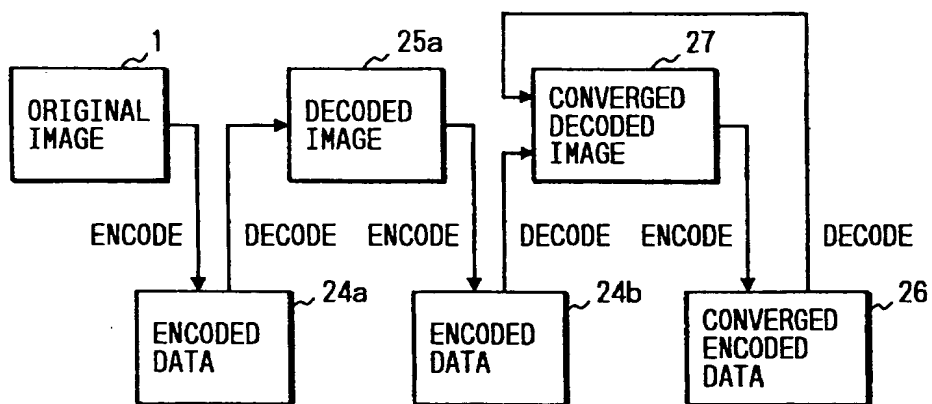


FIG. 3

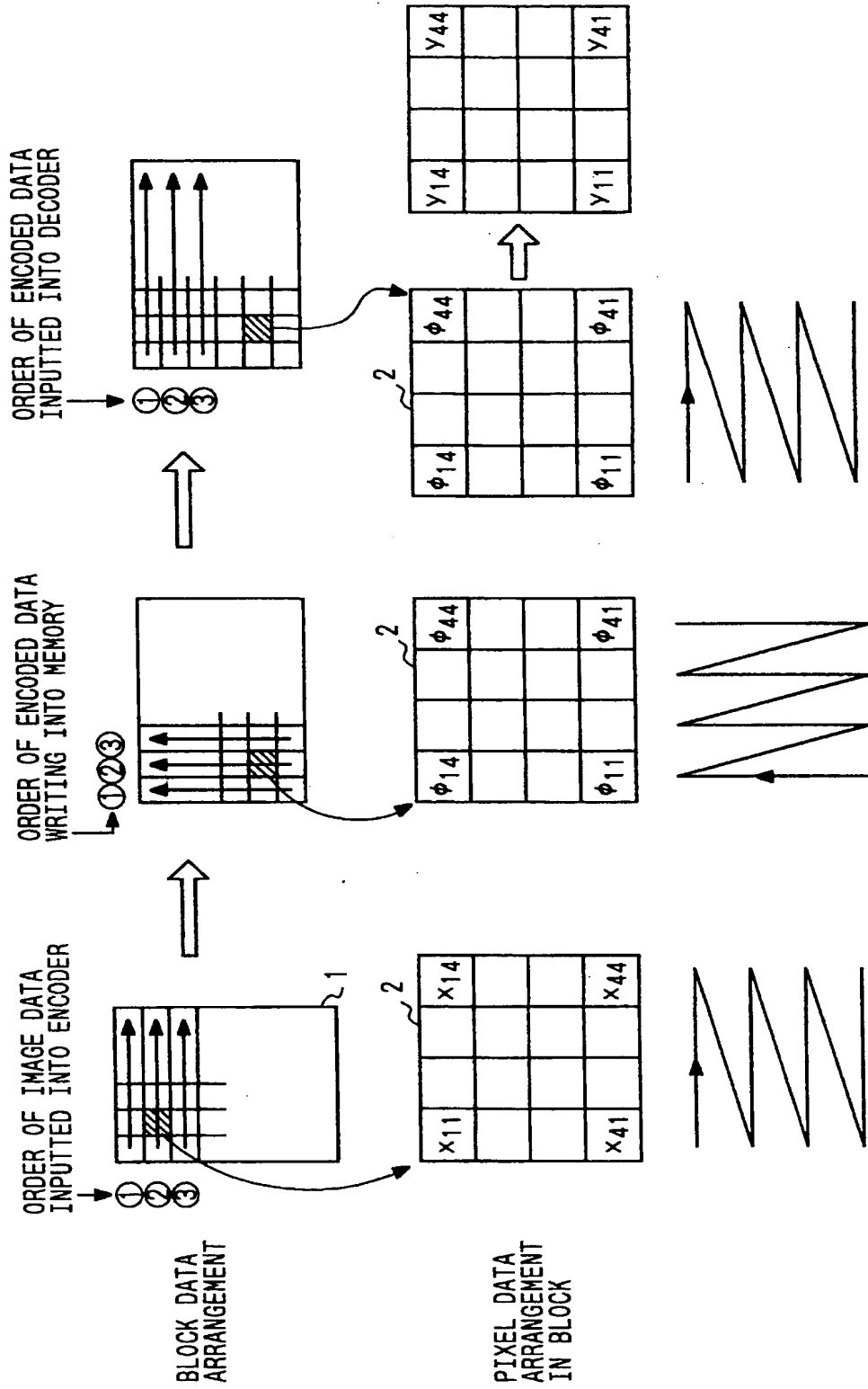


FIG. 4

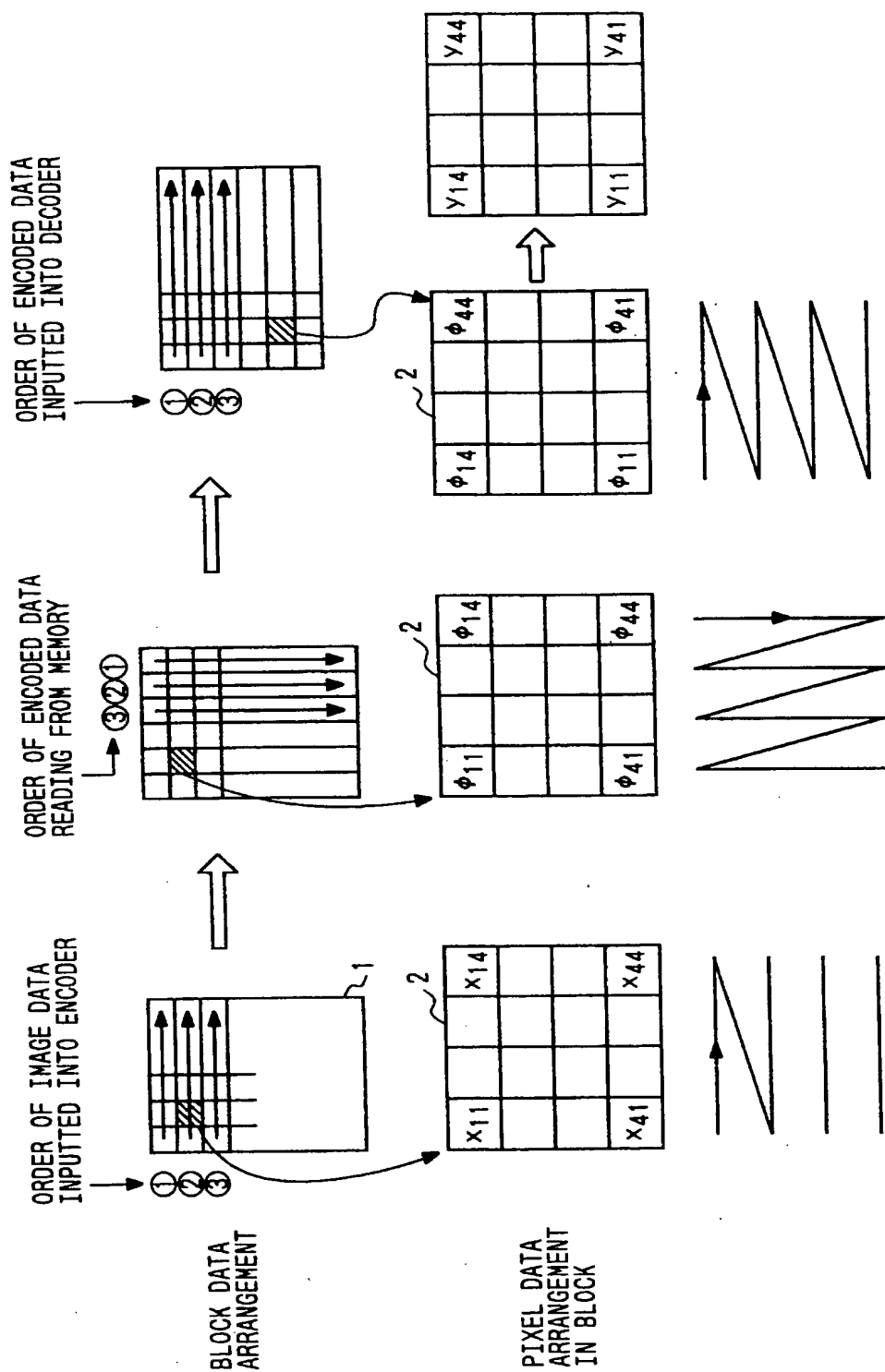


FIG. 5

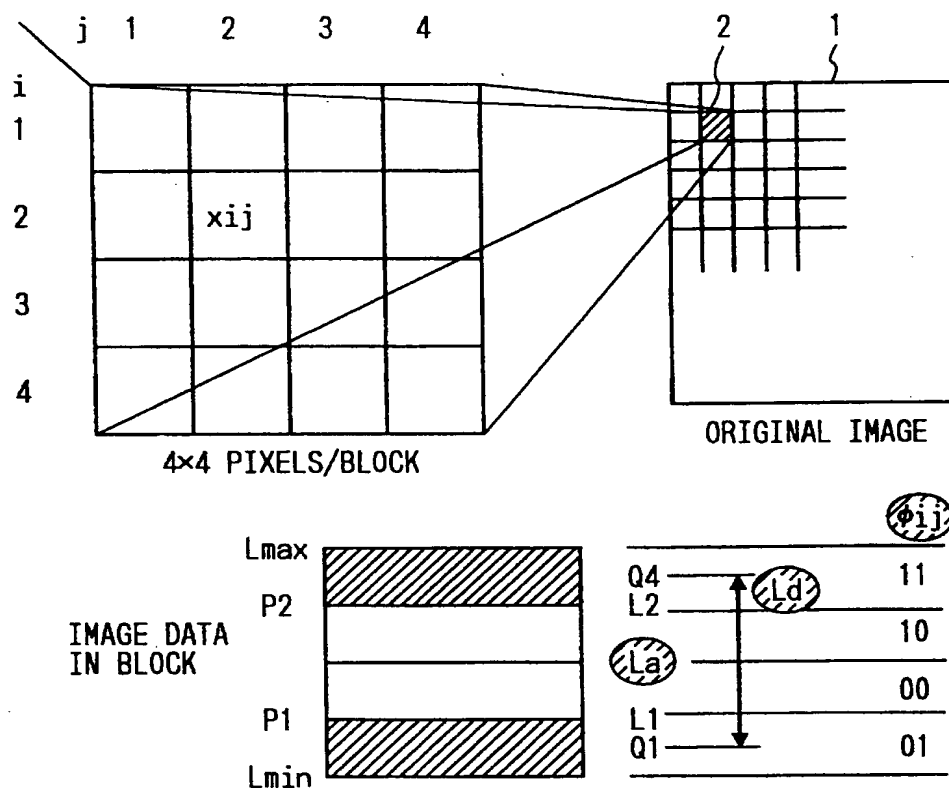


FIG. 6

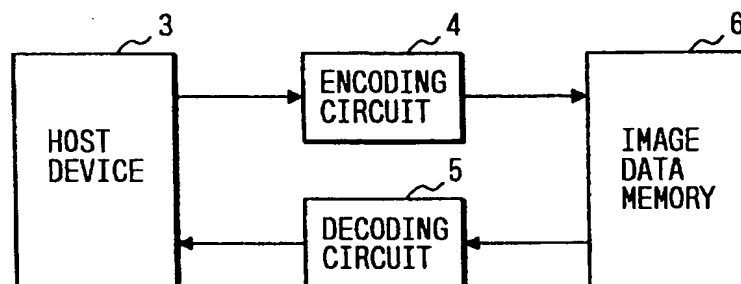


FIG. 7

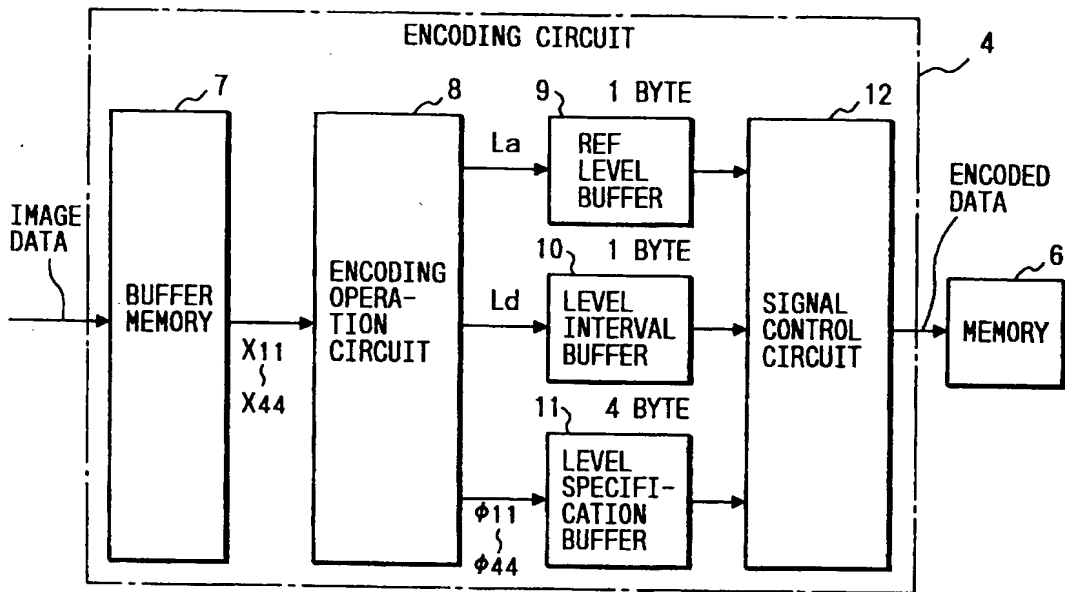


FIG. 8

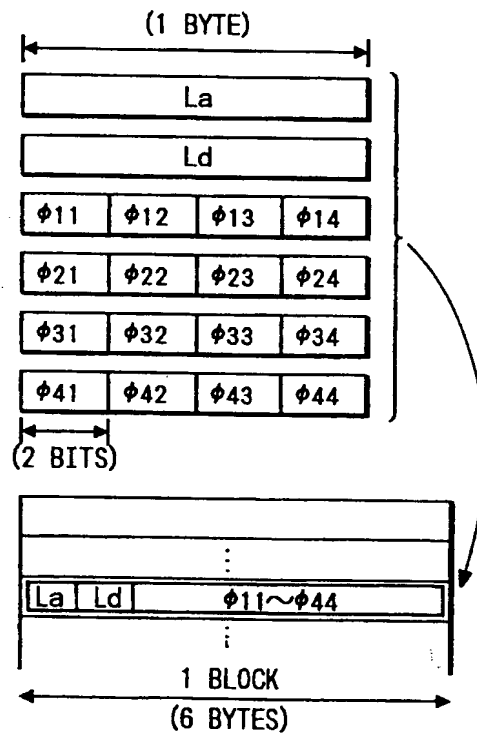


FIG. 9

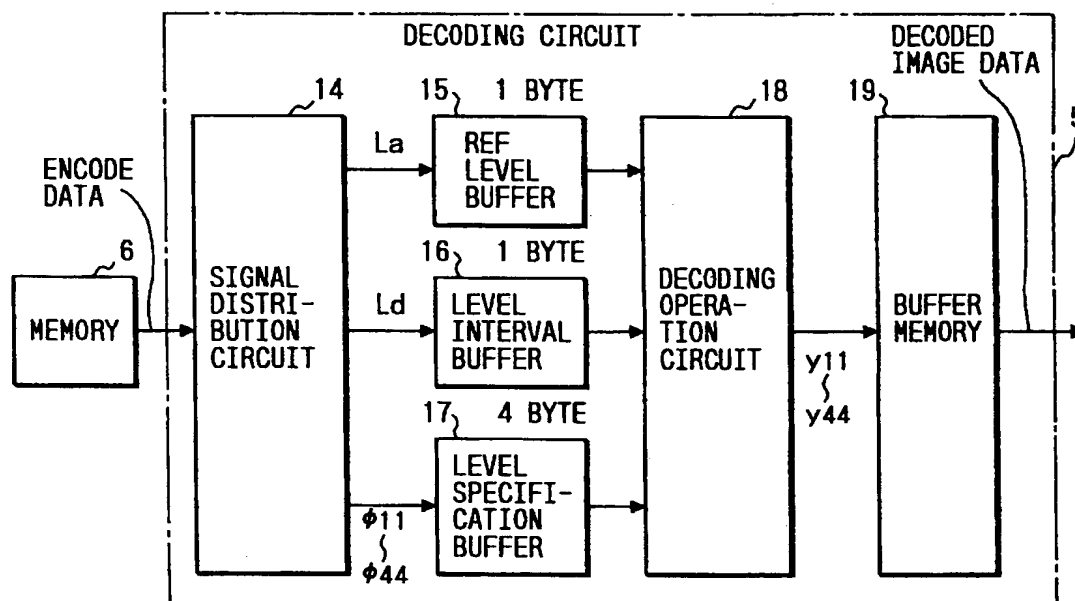
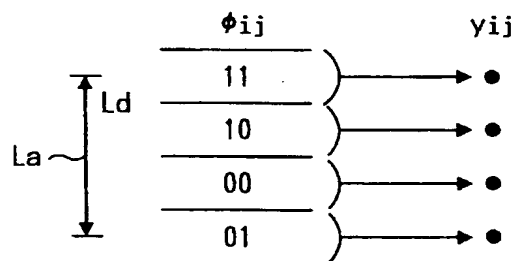


FIG. 10







European Patent  
Office

## EUROPEAN SEARCH REPORT

Application Number  
EP 93 11 3923

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (Int.CLS)
Y	EP-A-0 261 561 (FREDERIKSEN & SHU LABORATORIES, INC.) 30 March 1988	1,2	H04N1/41
A	* abstract; figures 2,3 * * column 6, line 36 - column 10, line 15 * ---	3,4	
D,Y	PATENT ABSTRACTS OF JAPAN vol. 13, no. 481 (E-838)31 October 1989 & JP-A-11 088 166 (NIPPON TELEGR & TELEPH CORP <NTT>) 27 July 1989	1,2	
A	* abstract * ---	3,4	
P,A	EP-A-0 549 309 (XEROX CORPORATION) 30 June 1993 * abstract * -----	3,4	
The present search report has been drawn up for all claims			TECHNICAL FIELDS SEARCHED (Int.CLS)
			H04N G06F
Place of search		Date of completion of the search	Examiner
THE HAGUE		16 February 1994	Revellio, H
CATEGORY OF CITED DOCUMENTS			
X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document		T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons ----- & : member of the same patent family, corresponding document	